A Machine Learning Approach to Residential Property Price Indexes

Felix Chan, Greg Costello, Rainer Schulz, and Zhuoran Zhang*

October 15, 2021

*Chan, Costello, and Zhang: School of Accounting, Economics and Finance, Curtin University, Building 408, Kent Street, Perth 6102, Australia; felix.chan@cbs.curtin.edu.au, greg.costello@cbs.curtin.edu.au, and zhuoran.zhang1@postgrad.curtin.edu.au. Schulz: University of Aberdeen Business School, Edward Wright Building, Dunbar Street, Aberdeen AB24 3QY, United Kingdom, r.schulz@abdn.ac.uk.

Abstract

Keywords: machine learning, missing values and outliers, residential property price index

JEL Classification: C14, C81, R31

1 Introduction

Research Question:

1. machine learning approach which could include missing values (Hill 2018)
2. in the past, RPPI in a large area and in submarkets need to calculated separately and process is inefficient. pdp could solve this one. (krause 2019, greg, goh 2014)
3. updating problems when machine learning applied in indexing field. (krause 2019, euro stat index handbook)

2 Data

2.1 Market and data

Perth is the capital and largest city of Western Australia, one of the five biggest city in Australia. There are around 2.1 million residents living in Great Perth area in 2020, the last year in our sample period from 2015M1 to 2020M12. Around $24,000$–$34,000$ properties are sold in metropolitan area each year, mostly houses (66.1%), followed by group houses (12.7%) and the rest types[1].

The transaction data set is provided by the *Western Australian Land Information Authority*, that operates under the business name of *Landgate*. The *Landgate* data contain all transaction records in Great Perth metropolitan area (excluding Mandurah) including vacant parcels and buildings. Each record documents information about sales price and date, parcel details, property

[1]The detailed property classes are shown in Appendix.

type, number of bedrooms, bathrooms and a range of other building features. The parcel details include the identity information of each property, such as the parcel ID, Unique polygon ID number (PIN), land ID and application number of transaction registration. These IDs are helpful to search and distinguish the transactions of each property. The coordinates are available in a separate cadastral file, linked to each property by PIN. From 2015 to 2020, there are 21,231 new established property sold[2]. The rest are old properties those had been sold at least once before 2015 or those are listed on market for the first time.

2.2 Data preparation

The *Landgate* data contains 185,980 transactions of residential properties from 2015 to 2020. Firstly, we exclude bundle sales, duplicate records, data errors and off-plan property sales, as they are not relevant and the removal does not obtain bias on the overall sample (**?**). In addition, non-market transaction records in *Landgate* data are eliminated. The yearly minimum and maximum prices of sold properties in Perth local housing market are set as boundaries of transaction prices from 2015 to 2020. The source of this information is *Australian Urban Research Infrastructure Network* (AURIN), the data provider is *Australian Property Monitors* (APM). Finally, there are 174,137 observations rest in our data set, 11,843 observations (6.3%) filtered in total. Hereinafter this data set is referred to as the 'raw data'. The eliminating process is summarized in **??**.

[**??** about here.]

[2]This number includes the buildings which are built after 2014.

The summary of variables in the prepared 'raw data' are presented in **??**[3]. Clearly, we face two problems when the 'raw data' are used to estimate the prices of residential properties. First one is the missing values shown in the essential characteristics. Most of essential characteristics show low level of missing values, the rates are higher than 0.09% but lower than 1.6%. The missing rate of floor area, however, is around 35%, that is much heavier than the others. The second is outlying observations in the 'raw data'. There are some unusual values in the Max column of **??**, such as 33 bedrooms and 21 study rooms. Missing values and outliers should be dealt, because they may affect the process and accuracy of estimation.

[**??** about here.]

3 Methodology

3.1 Gradient boosting machine

Gradient boosting machine (GBM, **??**) is another approach to improve the predictions resulting from a decision tree. GBM also plants a "forest" of trees as random forest. In random forest, each tree is grown on a bootstrapped data, independent from the other trees. In GBM, however, each tree is grown sequentially using information from previously grown trees on a fraction of training set (without replacement). The trees are in the same "family line". The GBM fitting procedure is also different from random forest. Firstly, it sets the predict values equal to zero, the residuals are equal to the value of

[3]The details about missing values and outliers are shown in appendix.

dependent variable of observations in the training set.

$$\hat{f}(X) = 0 \quad and \quad \varepsilon = Y - 0. \tag{1}$$

Then, the number of tree (N) and the number of splits[4] (d) are pre-decided. A tree ($\hat{f}^n(X)$) is fitted with d splits using the training data. The predict values are updated by adding in a shrunken version of the new tree.

$$\begin{aligned} \hat{f}(X) &\leftarrow \hat{f}(X) + \lambda \hat{f}^n(X). \\ \varepsilon &\leftarrow \varepsilon - \lambda \hat{f}^n(X). \end{aligned} \tag{2}$$

After repeating N times, the final output predict values are

$$\hat{f}(X) = \sum_{n=1}^{N} \lambda \hat{f}^n(X). \tag{3}$$

At this moment, the residuals are stable and minimized depending on current available training data set and tuning parameters. Differing from random forest's averaging process, GBM applies an updating algorithm.

"Missing branch" approach

Commonly, tree based models are under binary tree structure. A split decision partitions observations to the 'left' child node or the 'right' child node, this depends on the value of selected primary variable. However, it is impossible to assign a missing value. Adding a new branch on binary tree split could solve this situation. All instances with a missing value in primary variable will be assigned to the missing branch directly. The rest of non-missing attributes will be divided as usual. The observations in the 'missing' node could be partitioned again as long as the estimation would be improved. Missing branch is created in each split as security mechanism, in case that missing attributes

[4]Generally, the interaction depth is used, these two tuning parameter are similar.

appear only in test data. A schematic example is shown in **??**. If there are missing attributes in the primary variable (X_1) of the top split, the missings will be assigned to $R_{na,1} = \{X|X_1 = NA\}$. The rest will be distributed to the 'left' if $X_1 < c_1$, and to the 'right' if $X_1 \geq c_1$. If X_2 is unobserved in some instances, they will be assigned to $R_{na,2}$ or $R_{na,3}$.

[**??** about here.]

3.2 Interpretability: partial dependence plot

3.3 The analysis design

The estimation modules of AVM systems are selected from three model families, random forest, gradient boosting machine (GBM) and basic linear model. One estimation module cooperates with the modules introduced previously as one complete system. For the purpose of comparison, there are eleven combinations, three systems in each tree based machine learning family and two linear model benchmarks. In addition, we apply rolling windows strategy to study the prediction performance of modular AVM systems that mimics the updating process of property price evaluation in practice. The training data sets contain observations in two-year (eight-quarter) length period. The first training period is from the first quarter of 2015 to the forth quarter of 2016. We repeatedly shift this window by one quarter and re-apply the modular AVM systems. The testing data sets are constructed by instances in the following quarter of training periods. The evaluation metrics are calculated always on the testing data. The first testing sample is for the first quarter of 2017. In total, we have 16 different windows in the 'raw data', the fraction of missingness in each window is summarized in **??**.

[?? about here.]

3.3.1 Eight modular AVM systems

According to the cooperation of modules, modular AVM systems could be divided into two main groups, the bundle approaches and the stand-alone approaches. The bundle approaches are those the estimation modules could work with the other modules concurrently for solving the three-pronged problem (missing values, outliers and price estimation). The stand-alone approaches also rely on modules for solving missing and outlying values. But they deal with problems in a designed sequence. Modules need to deal with missing values or outliers firstly. Then, estimation modules are applied. All of modular AVM systems are implemented with the statistical software R, version 4.1.1 (?).

The bundle approaches

Model specifications of random forest ($R1$) and GBM ($G1$) are similar to those of standard hedonic price models, based on hedonic pricing theory (?). The response is the logarithm of transaction price and the predictors are features those could explain variance in the price:

$$log(p) = f(s, l, t) \tag{4}$$

where p is the transaction price, s are structural characteristics shown in ?? and property types, l are location features (latitude, longitude and LGAs) and t is temporal feature (continuous quarter number).

$R1$ applies on-the-fly imputation to deal with missing values[5]. In addition, each tree in random forest uses a bootstrapped set of full training set ("bagging"), which could reduce the effect of noises on each tree (**?**). GBM ($G1$) assigns observations, those have missing value in split variables, to the "missing branch". The assigned observations could be partitioned if the other variables are available. Similar to "bagging", GBM trees use random subsets of full training set[6], which could also reduce the effect of outiers on each tree. Due to that all modules could simultaneously run, $R1$ and $G1$ could be directly applied on training sets with no prerequisites.

The stand-alone approaches

The stand-alone approaches apply modules one by one, estimation modules are always the last. $R2$ & $R3$ and $G2$ & $G3$[7] respectively belong to random forest family and GBM family. They inherit the same model specification. The difference between the pairs is the path of system running. In $R2$ and $G2$, the missing values in each training data set are firstly imputed. The multiple imputation is applied, missing values are imputed five times and five training sets in one training period are generated. Then outliers in these five imputed

[5]Some literatures (see samples **??**) claim that random forest could apply surrogates. However, **?** suggest that surrogates may not be well suited to forests. Random forest randomly selects variables in each split, such that variables within a node may be uncorrelated, and a reasonable surrogate split may not exist. For this reason, we don't use random forest with surrogates to solve missing value issue.

[6]This process samples data without replacement. The difference between this process and "bagging" is whether the subsampling process is with replacement or not.

[7]In stand-alone approaches, random forest will switch off on-the-fly imputation and bagging. Similarly, GBM will turn off subsampling option. If training set has no missing values, missing branch will not be activated.

training sets are detected by isolation forest and removed. After these preparations done by modules, estimation modules take their responsibility on these prepared training sets in each rolling window. On the contrary, $R3$ and $G3$ run on an opposite path, removing outliers firstly and then imputing missing values. The benchmarks ($L1\&L2$) refer to the hedonic model specification in **?**. The model form may also summarized as

$$log(p) = f(s, l, t) \tag{5}$$

where p is the transaction price. s are structural characteristics and dummies of property types. One new structure variables, ratio of bathrooms to bedrooms, is added. Lounge, kitchen, family room, meal room and tennis court are excluded from the structure variable list. Because they are not chosen in the subset selection process through sixteen windows. l means location features (distance from CBD[8] and LGAs dummies) and t is temporal feature (quarter dummies).

Given these different modular AVM systems those are available to deal with missing and outlying values and do price estimation, **??** briefly depict the combinations of modules and their process sequence.

[**??** about here.]

3.3.2 Evaluation metrics

Different evaluation metrics could indicate different ranks of models, each metric could only assess one aspect of models. **?** comprehensively investigate the

[8]The distance between the location of the property and Perth CBD (city town hall) is calculated using coordinates.

metrics for evaluating the performance of AVMs. Depending on their recommendations and actual situation in our case, the assessment measures we use are summarized in **??**. They cover average bias, absolute ratio, squared ratio and percentage ratio, four major groups of metrics.

[**??** about here.]

In our study, we focus on root mean squared error (RMSE) and percentage error range (PER(a)). Both of them take bias and dispersion of errors into account. The PER(a) gives the fraction of errors that fall out of the interval $[-a, a]$. For instance, given $a = 10$ and we assume $PER(10) = 50$, this informs that there are half of the valuations those errors are larger than 10 percent of their market values. The RMSE assess bias and variance of errors simultaneously, and it generates more detailed decimals that are more easily compared across methods than mean squared error.

3.3.3 Robustness

When the preferred approach or the top tier of approaches are recommended, the robustness of this recommendation is considered. In order to simulating the uncertainty of future property stocks, 500 bootstrapped test sets are generated in each rolling window. The transaction prices are predicted by eleven modular AVM systems using these 500 bootstrapped test sets. Evaluation metrics are calculated, the range of them are provided in each rolling window. The idea of this process is that the observed transaction information used to build a modular AVM system is immutable. However, future transactions are uncertain. Bootstrapped test sets could simulate the uncertainty the systems

could face in future. The forecast performance ranges of modular AVM systems provide an evidence. We, then, may have high confidence to recommend systems according to their forecast accuracy of future transactions.

4 Empirical results

5 Conclusion

Acknowledgements

The usual disclaimer applies.